UNIVERSITY OF
ILLINOIS LIBRARY
AT URBANA-CHAMPAIGN
ENGINEERING

NOTICE: **Return or renew all Library Materials! The *Minimum* Fee for each Lost Book is $50.00.**

The person charging this material is responsible for its return to the library from which it was withdrawn on or before the **Latest Date** stamped below.

Theft, mutilation, and underlining of books are reasons for disciplinary action and may result in dismissal from the University.
To renew call Telephone Center, 333-8400

UNIVERSITY OF ILLINOIS LIBRARY AT URBANA-CHAMPAIGN

JUN 27 1989

ENGINEERING

NOV 25 1995

NOV 22 REC'D

L161—O-1096

510.84
Il63c
no.94

Engin.

ENGINEERING LIBRARY
UNIVERSITY OF ILLINOIS
URBANA, ILLINOIS

CONFERENCE ROOM

Center for Advanced Computation

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN
URBANA, ILLINOIS 61801

CAC Document No. 94

MATHEMATICAL SOFTWARE AND COMPUTER NETWORKS

by

Lawrence M. McDaniel

October 22, 1973

Digitized by the Internet Archive
in 2012 with funding from
University of Illinois Urbana-Champaign

http://archive.org/details/mathematicalsoft94mcda

CAC Document No. 94

MATHEMATICAL SOFTWARE AND COMPUTER NETWORKS

by

Lawrence M. McDaniel

Center for Advanced Computation
University of Illinois at Urbana-Champaign
Urbana, Illinois 61801

October 22, 1973

This work was supported in part by the Advanced Research Projects Agency of the Department of Defense and was monitored by the U. S. Army Research Office, Durham, North Carolina, under Contract No. DAHC04-72-C-0001.

TABLE OF CONTENTS

Mathematical Software and Computer Networks

The crucial problem of producing good quality mathematical software has finally been getting the attention it has so long deserved. A considerable amount of time and money is spent on the production of software which is sometimes poor in quality, and at best, a duplication of something somewhere else. In view of this poor state of affairs, the creation of a center to coordinate efforts in this area would be an excellent step in the right direction.

Recently, efforts have been directed at the development of high quality mathematical and statistical libraries, such as the Bell Laboratories Library One project, IMSL Library I from IMSL Inc., and EISPACK, produced by the NATS (NSF, Argonne, Texas, and Stanford--or the National Activity to Test Software) project. These projects have done well in addressing vital concerns in the creation of excellent software such as design, standards, validation, and documentation. As a result, users may now devote full attention to their specific problems, without questioning the credibility of the computer center's mathematical software.

Accessibility to software is often the stumbling block in providing a user community the benefits of an otherwise well-designed, decently standardized, validated, and excellently documented computer algorithm. The advent of computer networks has eased the inaccessibility problem significantly.

Furthermore, the capability of addressing several resources within a computer network enhances the utility of software. Two such resources on the ARPA Network have been integrated: EISPACK, the NATS library of Wilkinson's eigensystem routines, and SPEAKEASY, an interactive communication system.

Portability of Software

Portability, or the capability to utilize algorithms in variable environments, is of primary concern in the development of mathematical software. Constraints imposed by hardware, as well as by software, should be kept to a minimum, but unfortunately this is not always possible.

In some algorithms the smallest number representable on the computer is required, for example, ϵ, for which $1 + \epsilon > 1$. Such an accuracy factor is often a vital parameter in the performance of an algorithm, and must vary with the word length of the machine.

Some computers even have special built-in hardware features, such as double-precision accumulation of inner products on the KDF 9 (the machine primarily employed in the testing of Wilkinson's Linear Algebra routines).

The resemblance of a computer program to that of its corresponding algorithm is naturally a function of the selection of a programming language. Consider the following FORTRAN and corresponding ALGOL code:

DO 10 I = J,K	<u>for</u> i: = j <u>step</u> 1 <u>until</u> k <u>do</u>
. . .	<u>begin</u>
. . .	. . .
. . .	. . .
10 CONTINUE	<u>end</u> i;

In the case of $j > k$, the for loop in ALGOL would be bypassed; however, in the FORTRAN code, the DO loop would be executed exactly once. Constraints imposed by software as in the example above can be transcended with some minimal modifications to the computer program. However, trying to reprogram algorithms that exploit particular advantages of a programming language, for example string handling in PL/1, can prove quite difficult in some other language.

Attempts to create the most portable computer algorithm necessarily must minimize the impact of software and hardware incompatibilities. As a result some sacrifices in program design are inevitable. However, if it were possible to implement an algorithm in the language and for the machine it is best suited, and yet somehow make the algorithm simultaneously available to users of a variety of computer systems--the best of both worlds would certainly be realized!

Such a happy state of affairs becomes a reality in the realm of network computing, as is the case with the ARPA Network.

The ARPA Network

The ARPA (Advanced Research Projects Agency) Network is a system of computer resources designed to link ARPA-sponsored universities and research centers across the country. The ARPA Network itself is a full duplex high-speed (50,000 bits per second) data transmission network developed by Bolt, Beranek, and Newman of Cambridge, Massachusetts.

Nodes on the network are connected to each other directly or indirectly via sophisticated terminals called interface message processors (IMP). The heart of the IMP is a Honeywell DDP-516 computer which takes care of such tasks as error control, message routing, network tuning, and statistics gathering.

At any given node in the network one or more host computers may be attached to the IMP to provide a service center or research project with access to the network. While most of the host computers are associated with specific projects sponsored by ARPA, several locations are designated as service host sites. For instance, the 360/91 at UCLA is available to network users as a general computing service site. (See Figure 1 for the current ARPA Network logical map.)

Considered to be the leading candidate for becoming <u>the</u> nationwide data network, the ARPA Network offers diverse resources which make it unique among networks. Host machines include PDP-11s, 360/67, /75, /91s, PDP-10s, and ILLIAC IV. ARPA Network users are in the enviable position of selecting the computer that is best tailored to their individual needs.

The Center for Advanced Computation at the University of Illinois at Urbana-Champaign has been a primary network user for the past year. A PDP-11 based system known as ANTS (ARPA Network Terminal System) was developed at the center to provide researchers with easy terminal access to the network. This "minihost" computer system also provides the user with extensive peripheral capabilities such as sophisticated graphical and plotting devices as well as conventional card reading and printing facilities.

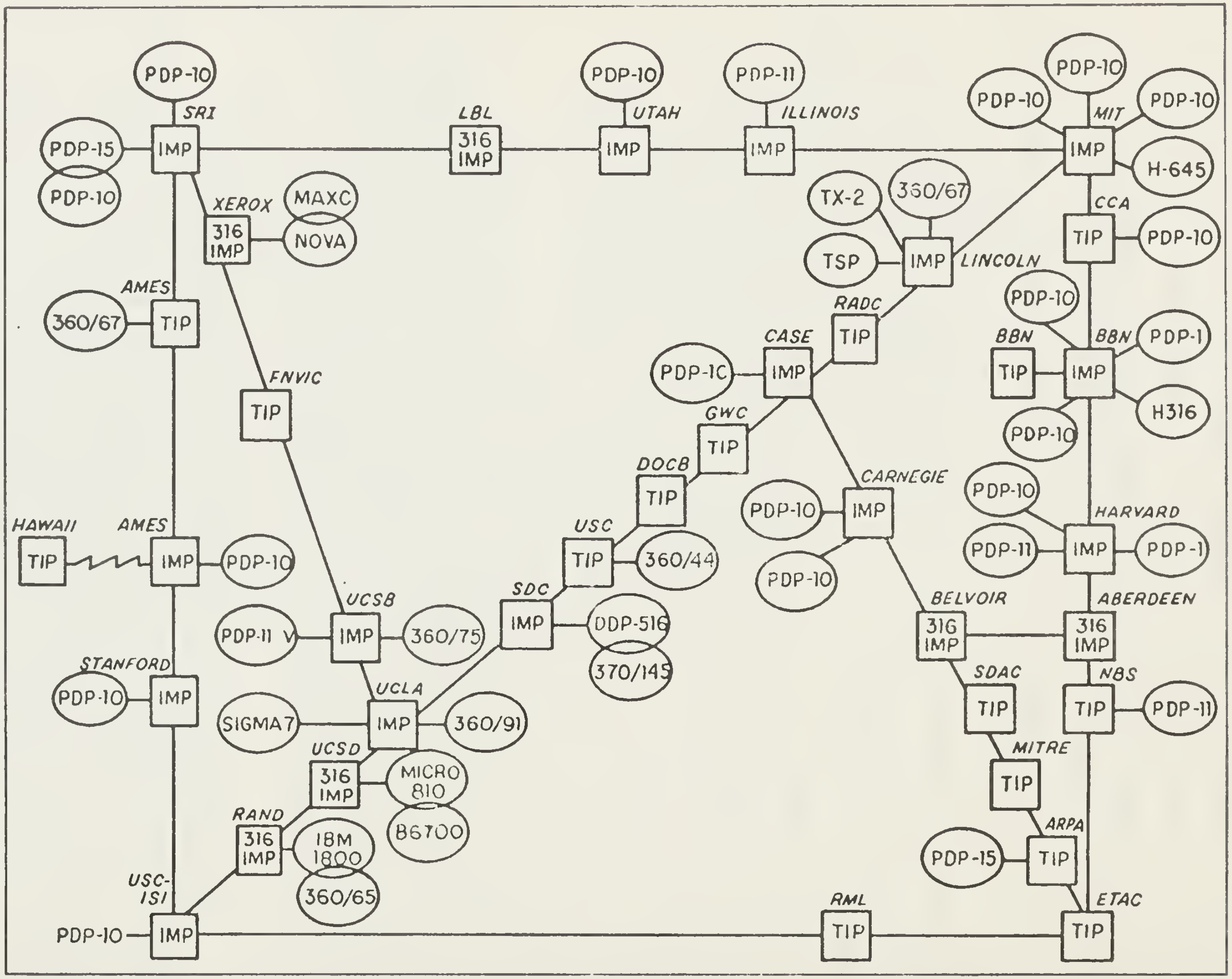

Figure 1. ARPA Network, Logical Map, May 1973

Involved in an extensive research program ranging from natural resource information systems and computer graphics to ILLIAC IV applications, the Center for Advanced Computation has relied extensively on the ARPA Network for its day-to-day computational requirements. Initially an experiment, the ARPA Network has demonstrated its utility and diversity and will probably be operated in the near future as a commercial enterprise. The use of networks in the world of computing is now a fact of life, and no doubt will have a profound effect on the trend of computing in the future.

EISPACK

EISPACK is an eigensystem package of FORTRAN subroutines produced by the NATS project. Most of the routines are translations of J. H. Wilkinson's ALGOL procedures from Volume II of the Handbook for Automatic Computation.

The library of EISPACK routines is unique in that an extensive certification process has been utilized to insure accurate performance of the codes. Approximately twenty test sites evaluated the package on IBM, CDC, PDP, Honeywell, and UNIVAC computer systems. An important portion of the package is the documentation accompanying the routines, which contains the following information on each routine:

1. Purpose
2. Usage
3. Discussion of method and algorithm
4. References
5. Checkout
6. Acknowledgements

The FORTRAN routines are well-designed, sprinkled generously with comments, and easily usable. The problem now is how does one gain access to a package of routines such as EISPACK?

The usual mode of communication is via the medium of magnetic tape. Upon receiving the tape, the installation makes the necessary adjustments to implement the package, which is a function of hardware and software incompatibilities. In the case of EISPACK, this boils down to anomalies in the job control language from installation to installation.

The entire procedure from the initial request of EISPACK to successful implementation takes about four to six weeks. This time period is naturally a function of the quality of the library, and the documentation accompanying the tape itself. One may not be so fortunate if he is dealing with a library that is not as portable as EISPACK.

EISPACK was successfully implemented on the 360/91 at the Campus Computing Network, UCLA. As previously mentioned the 360/91 provides the ARPA Network with a general computer facility. As of December 1972 approximately 125 EISPACK tapes had been requested and sent. Implementation of the package on the ARPA network automatically increased the number of sites possessing accessibility to EISPACK by 20 percent to 150. For these sites, no tapes had to be mailed, no additional waiting period was required, and, of course, the task of implementing the package was not required.

Information regarding the package was broadcast to all potential users of EISPACK via the ARPA Network Information Center (NIC). NIC provides ARPA Network users with a sophisticated information storage and retrieval system that is an integral part of any effective communication system.

A prospective EISPACK user could obtain the necessary information regarding the use of the package via this on-line information system. In addition any questions or problems he has with the package could be directed to the appropriate parties via the NIC system.

Finally, the whole problem of portability is alleviated. Algorithms need be implemented only once on the ARPA Network and may be accessed through a wide variety of computer systems. Designers of algorithms are not required to make sacrifices in the design of their codes in order to minimize hardware and software differences. By allowing computers to resolve incompatibilities among themselves, algorithms will be accessible to virtually any computer on the network and may be designed in a language and for the computer that best complement that algorithm.

SPEAKEASY

The ability of networks to virtually eliminate the problem of software portability has been recognized, but what about the problem of accessibility? In one sense networks can improve accessibility of software merely by opening appropriate channels of communication. However, even though the path to the desired software may be well defined, getting there is never half the fun!

The traditional means of access to an algorithm is via another computer program. The prospective user is placed in an error prone environment. As a result researchers find themselves spending entirely too much of their valuable time writing and debugging computer programs.

SPEAKEASY is a language that relieves users of the trivial tasks associated with writing conventional programs and enables conversation with a computer in a language natural for scientists. Exploratory calculations may be quickly formulated and carried to completion in SPEAKEASY, and as a result, utilizing the power and resources of modern large-scale computers becomes a pleasant experience for the researcher.

Consider the FORTRAN code required to evaluate the matrix product $C = AB^T$ where A and B are matrices of order m x n;

```
         DO  10 I = 1, M
            DO 10 J = 1, M
            C (I, J) = 0.0
               DO  10 K = 1, N
               C (I, J) = C (I, J) + A (I, K) * B (J, K)
      10 CONTINUE
```

This cumbersome combination of loops and subscripts is accomplished in SPEAKEASY by:

```
C = A * TRANSPOSE (B)
```

which closely resembles the original mathematical form.

A powerful and extensive vocabulary of commonly used operations is at the fingertips of the user. For example, the following statements calculate and display the inverse and eigenvalues of the indicated 3 x 3 matrix:

```
X = MATRIX (3,3 : 2,3,4,3,6,0,4,0,5)
PRINT X; 1/X; EIGENVALS(X)
```

SPEAKEASY would respond to the above statements with:

```
X (A 3 BY 3 MATRIX)
2  3  4
3  6  0
4  0  5

1/X (A 3 BY 3 MATRIX)
-.37037    .18519    .2963
 .18519    .074074  -.14815
 .2963    -.14815   -.037037

EIGENVALS(X) (A VECTOR WITH 3 COMPONENTS)
9          5.6056   -1.6056
```

A significant feature of the language is the capability of obtaining the complete SPEAKEASY vocabulary and information regarding the usage of any particular word.

The system responds to the statement

```
HELP MATRIX
```

with:

```
MATRIX(N,M:) defines an N-by-M matrix.  If no additional arguments
     are present, the matrix has all elements set to zero.
A shortened form is MAT.
MATRIX(N,M:I,J,...,K) defines an N-by-M matrix with preset elements.
     The elements are set row by row by use of the values I,J,...,K
     or the elements of I,J,...,K if they are structured objects.  If
     a complex element is encountered, then a complex matrix is
     defined.  If all the elements are not specified by the element
     list, the unspecified elements are set to zero.
```

The SPEAKEASY system was developed at Argonne National Laboratory under the direction of Dr. Stanley Cohen of the Physics Division. It has been successfully implemented at several installations around the country under OS/360 for operation on the IBM 360/370 series computers, and of particular note, SPEAKEASY is available on the 360/91 at UCLA to ARPA Network users.

A very important aspect of SPEAKEASY is the ease with which users may augment the vocabulary to meet their individual needs. One writes what is called a "linkule" in SPEAKEASY terminology, which provides

the interface between the SPEAKEASY processor and the routines that handle the actual computational tasks. Since linkule libraries are not part of the basic SPEAKEASY processor, users may utilize private linkules without having to worry about jeopardizing the integrity of the system.

The writing of linkules to access the EISPACK subroutines was a significant accomplishment because it demonstrated how a computer network can encourage resource sharing, and because it showed how accessibility to quality software may be greatly enhanced.

SPEAKEASY may be viewed as a communication system that permits users access to a wide variety of facilities in a computer complex, for example, data files, graphics and plotting equipment, and mathematical and statistical routines. This concept could naturally be extended into the realm of networks, so access to virtually any computer, and the resources of that computer, is made possible by a common communication system.

Conclusion

Consider now a network of heterogeneous computing systems linked to each other either directly or indirectly through intermediate nodes, and the existence of a well-designed, well-documented piece of mathematical software:

1. The algorithm having been implemented once, is now available to all nodes.

2. As the network is heterogeneous, programs residing on two diverse computing systems may utilize the same algorithm. These programs may, in fact, be written in languages different from that of the desired algorithm.

3. The designer of the algorithm may choose his language and machine to produce the most optimal algorithm, yet he is assured that his routine may be accessible to quite a diverse user community.

4. The algorithm can be linked to other network resources (in the manner that this paper has discussed the EISPACK/SPEAKEASY interface).

The ability of computer networks to encourage portability and accessibility of software by resource sharing is a promising trend in computing. Networks have opened channels of communication between hardware, software, and, most significantly, people that will enable more effective usage of computers in the future.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER CAC Document No. 94	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Mathematical Software and Computer Networks		5. TYPE OF REPORT & PERIOD COVERED Research Report - Interim
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Lawrence M. McDaniel		8. CONTRACT OR GRANT NUMBER(s) DAHC04-72-C-0001
9. PERFORMING ORGANIZATION NAME AND ADDRESS Center for Advanced Computation University of Illinois at Urbana-Champaign Urbana, Illinois 61801		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS ARPA Order No. 1899
11. CONTROLLING OFFICE NAME AND ADDRESS Advanced Research Projects Agency 1400 Wilson Boulevard Arlington, Virginia 22209		12. REPORT DATE October 22, 1973
		13. NUMBER OF PAGES 15
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) U. S. Army Research Office Duke Station Durham, North Carolina		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

No restriction on distribution.
Copies available from: National Technical Information Service
Springfield, Virginia 22151

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

ARPA Network
EISPACK
SPEAKEASY
Mathematical Software

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This paper discusses the impact of computer networks relative to the portability and accessibility of mathematical software. An application of resource sharing on computer networks is discussed -- the integration of two resources on the ARPA Network: SPEAKEASY, an interactive system for researchers and EISPACK, a library of eigensystem routines.

BIBLIOGRAPHIC DATA SHEET	1. Report No. UIUC-CAC-DN-73-94	2.	3. Recipient's Accession No.
4. Title and Subtitle MATHEMATICAL SOFTWARE AND COMPUTER NETWORKS			5. Report Date October 22, 1973 6.
7. Author(s) Lawrence M. McDaniel			8. Performing Organization Rept. No. CAC 94
9. Performing Organization Name and Address Center for Advanced Computation University of Illinois at Urbana-Champaign Urbana, Illinois 61801			10. Project/Task/Work Unit No. 11. Contract/Grant No. DAHC04-72-C-0001 ARPA Order No. 1899
12. Sponsoring Organization Name and Address Advanced Research Projects Agency as monitored by U. S. Army Research Office Duke Station Durham, North Carolina			13. Type of Report & Period Covered Research - interim 14.
15. Supplementary Notes None			

16. Abstracts

This paper discusses the impact of computer networks relative to the portability and accessibility of mathematical software. An application of resource sharing on computer networks is discussed -- the integration of two resources on the ARPA Network: SPEAKEASY, an interactive system for researchers and EISPACK, a library of eigensystem routines.

17. Key Words and Document Analysis. 17a. Descriptors

ARPA Network
EISPACK
SPEAKEASY
Mathematical Software

17b. Identifiers/Open-Ended Terms

17c. COSATI Field/Group

18. Availability Statement	19. Security Class (This Report)	21. No. of Pages
No restriction on distribution. Available from National Technical Information Service, Springfield, Virginia 22151	UNCLASSIFIED	13
	20. Security Class (This Page) UNCLASSIFIED	22. Price

BOUND BY
HNM

UNIVERSITY OF ILLINOIS-URBANA
510.84IL63C C001
CAC DOCUMENT$URBANA
91-98 1973-74
3 0112 007264283